

IN THE CLAIMS

Please amend the claims as follows:

Claims 1-10 (Cancelled).

Claim 11 (Currently Amended): A process for generating a performance model from a functional model for a system including a plurality of distributed hardware and software entities that engage to provide a service to at least one user, the process comprising:

distributing representative system requests in a finite number of groups and identifying, for each request group, a corresponding execution flow, the distributing of the requests being determined by a service being called upon and by characteristics of customer specific behavior, and the execution flow for each request group corresponding to a software entity execution linking, in sequence and/or in parallel, induced by a group request;

formalizing the execution flows using a notation allowing identification of causal relationships between different software entities of the system that are involved in the execution flows and allowing identification of information quantifying the system's resource consumption;

developing an intermediate model that comprises, in addition to the formalized execution flows, a resource specification that specifies physical hardware of the system, and an environment specification quantifying the amount of requests generated by said at least user; and

automating conversion of the developed intermediate model into a performance model.

Claim 12 (Previously Presented): The process according to claim 11, wherein the performance model derived from the developed intermediate model is dedicated to pre-existing software simulators using queuing network techniques.

Claim 13 (Previously Presented): The process according to claim 11, wherein the distributing the requests in a finite number of request groups is determined by the service being called upon, and by characteristics of the customer specific behavior that affect a way in which the service being called upon is delivered.

Claim 14 (Previously Presented): The process according to claim 11, wherein the execution flow for each request group is determined by the software entity execution linking, in sequence and/or in parallel, induced by a group request.

Claim 15 (Previously Presented): The process according to claim 14, wherein topology of a queuing model derived from the conversion is wholly determined by the execution flows corresponding to the request groups.

Claim 16 (Previously Presented): The process according to claim 14, wherein derivation of a performance model dedicated to a pre-existing simulator based on queuing network techniques can be automated by adapting correspondence rules proposed.

Claim 17 (Previously Presented): The process according to claim 16, wherein formalism of phases is achieved using an extension of a MSC (Message Sequence Charts) formalism.

Claim 18 (Previously Presented): The process according to claim 17, wherein the formalism of a graph of phases and execution flows of a service using a HMSC (High level Message Sequence Charts) formalism is represented in a form of a formalization tree comprising:

- a plurality of nodes representing phases constituting the service; and
- at least one oriented arc leading from a first node to a second node representing linking in a two-phase sequence.

Claim 19 (Previously Presented): The process according to claim 18, wherein the formalization tree further comprises:

- at least one node followed by plural arcs oriented in parallel; and
- at least one node followed by plural arcs oriented as a function of choice of a following phase depending either on a condition external to the system, or on an internal condition related to a current status of the system.

Claim 20 (Previously Presented): The process according to claim 19, wherein the intermediate model developed comprises the formalized execution flows characterizing the behavior of software entities and their interactions, at least one resource specification specifying the physical hardware, and at least one environment specification representing user behavior.